AMENDMENTS TO THE SPECIFICATION

Please replace paragraph [0039] with the following amended paragraph:

[0039] Historically, the temperature in thermal reactors was controlled by a spike TC control loop, using a PID control algorithm. By profiling the furnace in a static mode, using a paddle TC, the relation between paddle TC and spike TC under static conditions was established and stored in a profile table. Such a profiling procedure was performed at regular intervals or after maintenance. Because the paddle TC gives a more relevant reading for the actual wafer temperature, there has been a desire to use a paddle TC control loop that would make the time-consuming profiling procedure unnecessary. A control configuration 200 employing such a paddle TC control loop is shown in FIG. 2. In the configuration 200, an adder 210 computes an error signal $\mathbf{Es_{Pd}}$ from the paddle control setpoint *Pdset* and the actual paddle temperature *Pd*. Based on the error signal $\mathbf{Es_{Pd}}$, the PID controller 220 generates a power output signal **Pw** that is provided, via a thyristor unit, not shown, to the heating elements 230. The tube and wafers are indicated by 240. A feedback loop 250 that includes a digital filter [[251]] 260, provides the actual paddle TC signal to the adder 210. However, such a paddle TC control loop, as shown in FIG. 2, has such a strong non-linear behavior and long time constants that it is difficult or impossible to achieve a stable control loop with acceptable performance under dynamic conditions.

Please replace paragraph [0047] with the following amended paragraph:

[0047] FIG. 6 shows a hybrid control system [[501]] 600. The control system [[501]] 600 is an embodiment of the control system 500. In the control system [[501]] 600, the conventional controller 540 is based on a PID controller 542.

Please replace paragraph [0048] with the following amended paragraph:

[0048] Figure 7 shows a vertical thermal reactor system 700 where the vertical thermal reactor 100 is controlled by the hybrid cascade control system [[501]] 600. In the system 700, the process tube 110 is surrounded by the heating element 120, comprising multiple zone electric heating coils. Each zone has the spike TC 130 and the "profile" or paddle TC 140. The spike TC is located outside the process tube relatively near the heating element and the paddle TC is located inside the tube relatively near the wafers. A paddle control setpoint Pd_{set} and the actual paddle temperatures Pd are provided to an MBPC controller 720 (corresponding to an embodiment of the MBPC controller 520), which generates a spike control setpoint $\mathbf{Sp_{set}}$. An adder 730 computes a spike TC error signal using the spike control setpoint $\mathbf{Sp_{set}}$ and the actual spike temperatures, provided to the adder 730 via an inverter 732. A PID controller 740 uses the spike error signal to generate a power output signal that is provided to a power actuator 750 to provide power to control the heating element 120.

Please replace paragraph [0074] with the following amended paragraph:

[0074] Inputs to the MBPC controller 1200 are the paddle control setpoint temperature Pd_{set} and the actual paddle temperatures $\mathbf{P_d}$. The paddle control setpoint temperature is provided to the trajectory planning module 1220 and the actual paddle temperatures $\mathbf{P_d}$ are provided to a memory 1210 for storing past inputs and outputs. The memory 1210 provides input to the MBPC algorithm module 1230. Additional input for the models include actual spike temperatures. The Trajectory planning module 1220 generates N paddle control setpoints $\mathbf{Pd_{set}(1...}N\mathbf{)}$ distributed over a predictive horizon, where $\mathbf{Pd_{set}(1)}$ is the control setpoint for the present moment and $\mathbf{Pd_{set}(}N\mathbf{)}$ is the most future predicted control setpoint. These control setpoints $\mathbf{Pd_{set}(1...}N\mathbf{)}$ are provided to a first input of an adder [[1220]] 1222 via a line 1221. Further, the modeled paddle values $\widetilde{P}d_{fr}(1...N)$, which are provided as output by the MBPC control algorithm module 1230, are provided to a second input of the adder 1222 via a line 1233. The adder [[1220] 1222 calculates error signals Es(1...N) which are provided to the Optimizer module 1232 of the MBPC algorithm module 1230 via a line 1223. The optimizer module 1232

optimizes the model output by minimizing a cost function 1235 as represented by equation (35), using constraints 1236. The least-squares error between the modeled predicted paddle control setpoint temperatures $\widetilde{Pd}_{fr}(1...N)$ and the actual paddle control setpoint temperatures $\mathbf{Pd_{set}(1...N)}$ from the trajectory planner 1220 is minimized over the predictive horizon. The predicted paddle control setpoint temperatures are optimized by using the disturbance model (the last term in equation (35)) so that the predictive values approach the actual values.

Please replace paragraph [0075] with the following amended paragraph:

[0075] The spike correction value $\mathbf{\Delta Sp}$ is calculated, according to equation (45). The modeled values $\widetilde{Pd}_{fr}(1...N)$ are provided to the memory 1210 via a line 1234. The spike correction value $\boldsymbol{\Delta Sp}$ is provided from the MBPC algorithm into a spike output calculation module 1212 to calculate the modeled spike control setpoint $\boldsymbol{Sp_{set}(1)}$ according to equation (46). The modeled spike control setpoint $\mathbf{Sp_{set}(1)}$ is provided to the MBPC outputs fuzzy inference module 1240 via a line 1211. The modeled spike control setpoint value is provided to the Output limiter 1250 where the output is limited according to equation (54). The algorithms will be discussed in further detail below.

Please replace paragraph [0089] with the following amended paragraph:

[0089] The structure of a PID controller 1300 is shown in Figure 13 and reflects the structure of Equation (55). In the PID controller 1300, an error signal *Es* is provided to a limiter 1302, mathematically represented by equation (57) below. After passing through the limiter 1302, the limited error signal is provided to a differentiating action module 1320 via a line 1312, to an integrating action module 1340 via a line 1314, and to a proportional action module 1360 via a line 1316. In the differentiating module 1320, the rate of error signal change is determined in block 1322. Then the differentiating action is calculated in block 1324 using a $\mathbf{k_d}$ value, which is variable and calculated in block 1328. The calculated differentiating action passes the output limiter 1326, mathematically represented by Equation (59).

Please replace paragraph [0090] with the following amended paragraph:

[0090] In the integrating action module 1340, an integration constant $\mathbf{k_i}$ is calculated in block 1348 and applied in block 1342. A summation is carried out in block 1344. Then the calculated integrating action passes through the output limiter 1346, mathematically represented by Equation (58). In one embodiment, the output of the differentiating action module, *Dout*, is also used as an input for calculating the integrating action, as shown by feedback line 1331. The purpose of this feedback is to achieve improved control during ramp-up and ramp-down. The outputs of both the differentiating action module 1320 and the integrating action module 1340 are provided to the proportional action module 1360 via lines 1330 and 1350, respectively. In the proportional action module 1360, the proportionality action is calculated in block 1362, using a proportionality constant $\mathbf{k_p}$ calculated in block 1366. Then the output signal **Pw** passes through output limiter 1364, represented by Equation (60).

Please replace paragraph [0091] with the following amended paragraph:

[0091] The PID parameters $\mathbf{k_d}$, $\mathbf{k_i}$ and $\mathbf{k_p}$ are calculated in blocks 1328, 1348 and [[1368]] 1366 according to the formulas:

$$\begin{aligned}
k_d(t) &= \frac{r_p(t)T_{ds}(t)}{T_s} & & \\
k_i(t) &= k_{i_\max} - k_{i_\min} \times (T_{\max} - Sp_{set}(t)) & \quad T_{\min} \le Sp_{set}(t) \le T_{\max} & \quad (56)\\
k_p(t) &= k_{p_\max} - k_{p_\min} \times (T_{\max} - Sp_{set}(t)) & \quad T_{\min} \le Sp_{set}(t) \le T_{\max} &
\end{aligned}$$

where:

$$\begin{aligned}
k_{p_\max} &= k_{p0} + k_{p1} \times (\frac{Sp_n(T_{\max})}{T_{\max}} - k_{p2}) \\
k_{p_\min} &= \frac{k_{p_\max} - (k_{p0} + k_{p1} \times (\frac{Sp_n(T_{\min})}{T_{\min}} - k_{p2}))}{T_{\max} - T_{\min}}
\end{aligned}$$

$$k_{i\_max} = k_{i0} + k_{i1} \times (\frac{Sp_n(T_{max})}{T_{max}} - k_{i2})$$

$$k_{i\_min} = \frac{k_{i\_max} - (k_{i0} + k_{i1} \times (\frac{Sp_n(T_{min})}{T_{min}} - k_{i2}))}{T_{max} - T_{min}}$$

where $Sp_{set}(T)$ is calculated by using the static model according to Equation (3), and k_{p0}, k_{p1}, k_{p2}, k_{i0}, k_{i1}, and k_{i2} are predetermined constants relating to system gain and time constants. *Tds* is a delay time. T_{min} and T_{max} are the lower and upper temperature boundaries of the temperature control range. After dynamic and static models are identified, the k_{p_max}, k_{p_min}, k_{i_max}, k_{i_min} and *Tds* can be re-determined and modified by performing an analysis of both the dynamic and static models. For a given thermal reactor, these parameters can be predetermined in the design and development phase without additional on-line tuning for individual reactors after manufacturing or during use.

Please replace paragraph [0093] with the following amended paragraph:

[0093] By using Equations (57)-(60), the dynamic response of the inner-loop PID is stable and provides the desired speed of response. The control results are shown in FIG. 17. The model used was derived ~~can be seen~~ from modeling data sets in FIG. 9 and FIG. 10. The design of the outer loop MBPC based on the dynamic models derived from the closed-loop data controlled by the PID controller greatly increases the temperature system control margins and stability[[.]] as can be observed by comparing FIGS. 10 and 17. Although in both figures the inner spike loop has been active, in FIG. 10 the outer loop is "open" or inactive and in FIG. 17 the outer loop is closed, or active.